

Infinite Continuous Adaptivity for Incompressible SPH の実装

渡辺 拓希^{1,a)}

概要：筆者は SIGGRAPH 2017 で発表された論文 “Infinite Continuous Adaptivity for Incompressible SPH” の実装を行っている。この論文では、適応的な粒子サイズの設定を行う非圧縮性 Smoothed Particle Hydrodynamics (SPH) シミュレーションに関する新たな手法を提案しており、液体表面から内部に向かって連続的に粒子サイズを変化させることができる。手法の中心となる考え方は、液体表面までの距離に対する最適な粒子質量を定義し、近傍粒子との質量分配によって、最終的に定まった質量に応じた粒子サイズに変化させるというものである。ユーザ自身が粒子サイズを定義する従来法とは異なり、質量に応じて自動的に決まるため、粒子サイズは理論上無限段階に変化させることが可能となる。この手法を液体シミュレーションに適用することで、鮮明な表面形状を保ちながら、質量保存、数値的安定性、非圧縮性、そして粒子数削減による計算コストの軽減を可能にする。

Implementation of Infinite Continuous Adaptivity for Incompressible SPH

HIROKI WATANABE^{1,a)}

1. はじめに

粒子を用いたシミュレーション手法の一種である SPH 法は、物理シミュレーションの分野において様々な現象の再現に応用されている。従来の SPH 法は見た目の鮮明さを向上させるために、計算コストと引き換えに大量の粒子を用いる必要があるため、それに対し、表面付近の粒子は細かく、内部の粒子は粗くするといったような、空間的に粒子サイズを変更する適応性を考慮した手法が提案されてきた [2], [3]。これらの手法は、ユーザ自身がとびとびの粒子サイズを定義するため、滑らかな粒子サイズの変化が困難であり、またサイズ比率を大きくするとシミュレーションの安定性にも影響を及ぼしてしまう。

筆者が実装を行っている手法 [4] では、液体表面までの距離に対する最適な粒子質量の定義という考えのもと、近傍粒子との質量分配により定まった最終的な質量に応じて

粒子サイズを決定するため、ユーザがとびとびのサイズを定義することなく連続的なサイズ変化を実現する。これにより、従来法では困難であった滑らかな粒子サイズの変化およびサイズ比率をより大きくすることが可能となる。また、非圧縮性 SPH 法と組み合わせることで、より安定したシミュレーションを実現する。

2. 手法

適応的な SPH 法において、近傍粒子探索を行う際に用いる粒子 i の有効半径 h_i は式 (1) により計算される。

$$h_i = \eta \left(\frac{m_i}{\rho_i} \right)^{\frac{1}{3}} \quad (1)$$

ここで、 m は粒子の質量、 ρ は密度、 η はユーザが定義する定数である。有効半径が大きいほど粒子サイズも大きくなるが、非圧縮性により密度は変化しないことを考慮し、有効半径は質量のみに依存するパラメータとして扱う。

¹ 筑波大学大学院 図書館情報メディア研究科
〒305-8550 茨城県つくば市春日 1-2

^{a)} watanabeh@slis.tsukuba.ac.jp

2.1 最適な粒子質量の定義

各粒子の最適な粒子質量 m^{opt} は以下の式 (2) で定義される。

$$m_i^{opt} = m^{base} \left(\frac{\min(|\phi_i|, |\phi_{max}|)}{|\phi_{max}|} (1 - \alpha) + \alpha \right) \quad (2)$$

ここで、 ϕ は表面からの距離、 ϕ_{max} は最大距離、 $\alpha = m^{fine}/m^{base}$ 、 m^{fine} は最小質量、 m^{base} は $\phi_i = \phi_{max}$ のときの質量 (最大質量) である。表面からの距離は [2] を用いて計算する。

2.2 質量分配

各粒子において $m_i = m_i^{opt}$ となるように、近傍粒子との適切な質量分配を行う。分配方法は、式 (3) で決められる粒子の大きさのレベル C_i によって Split, Merge, Redistribution, Unchange のいずれかを適用する。

$$C_i = \begin{cases} S & m_i^{rel} < 0.5 \\ s & 0.5 \leq m_i^{rel} \leq 0.9 \\ o & 0.9 < m_i^{rel} < 1.1 \\ l & 1.1 \leq m_i^{rel} \leq 2.0 \\ L & 2.0 < m_i^{rel} \end{cases} \quad (3)$$

ここで、 $m_i^{rel} = m_i/m_i^{opt}$ である。Split は $C_i = L$ の粒子に適用し、1 個の粒子を m_i/m_i^{opt} 個に分割する。Merge は $C_i = S$ の粒子に適用し、近傍にある n 個の粒子 s および S に対して m_i/n ずつ質量を振り分けた後、粒子 i を削除する。Redistribution は $C_i = l$ の粒子に適用し、近傍にある n 個の粒子 s に対して $(m_i - m_i^{opt})/n$ ずつ質量を振り分ける。Unchange は $C_i = o$ の粒子に適用し、分配処理は行わない。また、一度分配処理に関わった粒子は、そのステップの終わりまで他の粒子による質量分配に関わらないようにすることで、分配処理が過剰に行われることを防ぐ。

2.3 密度および速度のブレンディング

Split または Merge により発生した粒子については、分配処理前の粒子の値を用いて密度および速度のブレンディングを行い、シミュレーションの安定化を図る。分配処理後に発生した粒子の密度 $\rho_i^{blended}$ と速度 $\mathbf{v}_i^{blended}$ はそれぞれ、

$$\rho_i^{blended} = (1 - \beta_i)\rho_i + \beta_i\rho_o \quad (4)$$

$$\mathbf{v}_i^{blended} = (1 - \beta_i)\mathbf{v}_i + \beta_i\mathbf{v}_o \quad (5)$$

と定義される。 β はブレンディングの重み、 ρ_o は分配処理前の粒子の位置における密度、 \mathbf{v}_o は分配処理後に発生した粒子の平均速度である。 β は Split によって発生した粒子の場合は 0.5, Merge の場合は 0.2 を初期値とし、シミュレーションが進むごとに 0.1 ずつ減らす ($\Delta\beta = -0.1$)。 $\beta > 0$ の間はいずれの分配処理にも参加せず、 $\beta = 0$ となるまでブレンディングを行う。

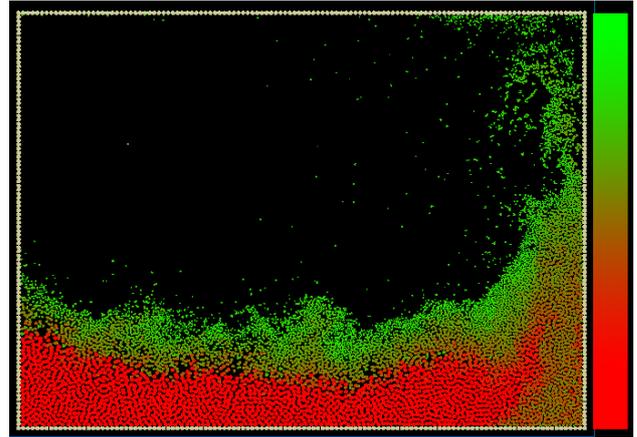


図 1 2D シミュレーションの実装

2.4 非圧縮性

非圧縮性を保つ手法として、論文では Implicit Incompressible SPH (IISPH) [1] を採用している。しかし、この手法は粒子質量および有効半径が不変であることを前提としているため、それらが変化することを考慮した式を新たに定義する必要がある。以下に、IISPH において粒子質量および有効半径が変化する場合の密度変化率の式を示す。

$$\frac{d\rho_i}{dt} = \frac{1}{\Omega_i} \sum_j m_j (\mathbf{v}_i - \mathbf{v}_j) \nabla W \quad (6)$$

ここで、

$$\Omega_i = \begin{cases} 1 + \frac{h_i}{3\rho_i} m_i \frac{\partial W}{\partial h} & C_i^{t-1} = l \\ 1 + \frac{h_i}{3\rho_i} \sum_j m_j \frac{\partial W}{\partial h} & (otherwise) \end{cases} \quad (7)$$

である。 j は近傍粒子のインデックス、 C_i^{t-1} は 1 ステップ前の粒子 i のレベル、 W は粒子間距離に依存する重み関数 (カーネル) である。また、圧力による力の計算は式 (8) を用いて行う。

$$\mathbf{F}_i^p = \sum_j \frac{m_j}{\rho_j} \left(\frac{p_i}{\rho_i^2 \Omega_i} + \frac{p_j}{\rho_j^2 \Omega_j} \right) \nabla W \quad (8)$$

3. 実装結果

現段階では、2D シミュレーションの実装まで終えている。図 1 は質量比を 10:1 とした場合のシミュレーション結果であり、表面から内部に向かって徐々に粒子サイズが大きくなっていることがわかる。シミュレーションの実装は C++ で行い、OpenGL を用いて粒子を描画している。

4. おわりに

実装にあたって、Split 後の粒子配置は特に調整が難しい部分であると感じた。分割数は Split 処理が適用された粒子によって異なるため、分割数によって適切な配置パターンを見つけなければ、安定したシミュレーションを行うことは難しくなる。実際図 1 の結果においても、Split 時に粒子がはじけるような挙動が時折確認された。今後は研究

会までにそれらの問題を改善し、シミュレーションの安定性を向上させるとともに、3D のシミュレーション結果を作成する予定である。

参考文献

- [1] Ihmsen, M., Cornelis, J., Solenthaler, B., Horvath, C. and Teschner, M.: Implicit incompressible SPH, *IEEE transactions on visualization and computer graphics*, Vol. 20 (2013).
- [2] Jon Horvath, C. and Solenthaler, B.: Mass Preserving Multi-Scale SPH Pixar Technical Memo 13-04, Technical report (2013).
- [3] Orthmann, J. and Kolb, A.: Temporal Blending for Adaptive SPH, *Computer Graphics Forum*, Vol. 31, pp. 2436–2449 (2012).
- [4] Winchenbach, R., Hochstetter, H. and Kolb, A.: Infinite Continuous Adaptivity for Incompressible SPH, Vol. 36, No. 4, New York, NY, USA, ACM, pp. 102:1–102:10 (online), DOI: 10.1145/3072959.3073713 (2017).